Energy, Mines and Resources Canada | Énergie, Mines et Ressources Canada

CANMET

Canada Centre for Mineral and Energy Technology | Centre canadien de la technologie des minéraux et de l'énergie

INTERFACING OF THE PYE UNICAM PU8610 UV/VIS SPECTROPHOTOMETER WITH THE OSBORNE 01 MICROCOMPUTER

P.L. Bedard, P. Richer, and K.G. Tan
Metallurgical Chemistry Section
Extractive Metallurgy Laboratory

December 1986

Project: 3.0.52.02
Chloride Processes for Sulphide Ore Treatment

This document is an unedited interim report prepared primarily for limited distribution to [illegible] for [illegible] (CANMET)

Ce document est un rapport [illegible] rédigé [illegible] de [illegible] nulle- [illegible] de [illegible] Centre canadien de [illegible] des minéraux et de [illegible] (CANMET)

40 pp

MINERAL SCIENCES LABORATORIES
DIVISION REPORT MSL 86-170 (TR)

MICROMEDIA

Crown Copyrights Reserved

INTERFACING OF THE PYE UNICAM PU8610 UV/VIS SPECTROPHOTOMETER WITH THE OSBORNE 01 MICROCOMPUTER

by

P.L. Bedard*, P. Richer*, and K.G. Tan**

ABSTRACT

Interfacing with a microcomputer was required to extend the versatility of the PU8610 spectrophotometer. Software programs were developed for the Osborne 01 that allowed remote control of the PU8610 in a menu driven mode. As a result, a very compact and inexpensive scanning UV/VIS instrument with data logging and data manipulation capabilities was obtained. If additional parts and machine codes can be found, a completely automatic on-line analyzer for concentrated hydrometallurgical process solutions may be developed. Description of the software and the logistics of interfacing are provided in this report as an aid for other users, and for future reference purposes.

---

*Technician, and **Research Scientist, Mineral Sciences Laboratories, CANMET, Energy, Mines and Resources Canada, 555 Booth Street, Ottawa, Ontario K1A 0G1.

MISE EN RELATION D'UN SPECTROPHOTOMÈTRE PYE UNICAM PU8610 UV/VIS AVEC UN MICRO-ORDINATEUR OSBORNE 01

par

P.L. Bédard*, P. Richer* et K.G. Tan**

RÉSUMÉ

Pour améliorer la versatilité du spectrophotomètre PU8610, il a été nécessaire de le relier à un micro-ordinateur. Des logiciels ont été développés pour l'Osborne 01, ce qui a permis un contrôle à distance du PU8610 par le biais d'un mode menu. C'est ainsi que fut obtenu un instrument à balayage UV/VIS compact et peu dispendieux, permettant l'enregistrement et la manipulation des données. Si des parties additionnelles et des codes machines sont trouvés, un analyseur interactif complètement automatique pourra être développé pour des solutions aux processus hydrométallurgiques concentrés. Une description du logiciel et de la logistique de cette interface est fournie dans ce rapport à titre d'aide aux usagés et dans un but de référence future.

---

*Technicien et **Chercheur scientifique, Laboratoires des Sciences minérales, CANMET, Energie, mines et Ressources Canada, 555 rue Booth, Ottawa, Ontario K1A 0G1

INTRODUCTION

Current efforts to modernize existing laboratory equipment are confronted by a spectrum of systems which are seldom compatible with one another in an electronic sense. The rapid development of new and improved instrumentation often renders yesterday's "best buy" as an inferior choice today. The only sensible and economic method to overcome these problems is to develop a know-how about the internal architecture of available instruments, and to incorporate improvements by means of software programming. Unfortunately, the competitive market for instrumentation and the high development costs for customized software often preclude the availability of architectural details and internal machine codes.

Recently, a PU8610 UV/VIS spectrophotometer was purchased to replace two elaborate instruments with electro-mechanical problems. The PU8610 is one of a new breed of compact microprocessor controlled instruments, and offers a wide range of built-in accessories to provide a rugged and versatile stand-alone tool for chemical analysis. Other, more sophisticated instruments were available, but these suffered from greater complexity, larger dimensions, and much higher price tags. Based on the technical specifications and informational details available, it was deemed that the PU8610 could be developed to match or exceed the state of the art by interfacing to an external microprocessor.

THE PU8610 UV/VIS SPECTROPHOTOMETER

The PU8600 line of Philips Electronics Ltd. (Pye Unicam in Canada) consists of compact, microprocessor controlled, single-beam spectrophotometers operating in the ultraviolet, visible and near infrared regions,

with or without built-in printers, automatic sample changers, autocells (flow-through sippers), electronic interfaces, and temperature controllers. Their immediate strengths lie in their rugged portability and the relatively low price for such features as routine programmability, autocalibration, synchronized sampling and recording, peak maximization, kinetic derivations and, especially, the ease of operation. The inherent shortcomings of single beam optics are compensated for by fast and reliable detection, signal averaging, electronic stabilization, and diagnostic autotesting. Performance data are superior to conventional instruments, and are augmented by silica coated optics, master holographic gratings, and by audio-visual fault warnings.

Technical and operational details are adequately documented in several publications (1-4), and a cursory description of some relevant features will suffice. The touch button control panel consists of three parameter-selection columns, a digital key pad, and a few additional controls. The PU8610 version includes all built-in options and offers five basic modes of operation: absorption, concentration standard, concentration factor, transmission, and kinetics. Parameters to be entered are the temperature, wavelength, program number, concentration factor, time delay (for equilibration), time of measurement (for noise averaging), cycles (repeat measurements), sample volume (for autocell), and additional tubing/ pump calibration factors. Additional controls for programming, calibration, peak optimization, and sample disposal also are provided.

An abbreviated summary (3) of the internal machine commands is listed in Table 1. Combinations of up to 3 codes permit access to a range of secondary functions, and a list of error messages is available (2). The fundamentals for using the RS232C interface are described (3), but examples

are given for the HP85 microcomputer only and the corresponding software packages were offered for about $2000 for a complete set. Since access only to CP/M based microcomputers was available, the appropriate software for remote control of the PU8610 had to be developed.

THE OSBORNE 01 MICROCOMPUTER

The Osborne 01 microcomputer was selected to match the portability and the ruggedness of the PU8610. A complete description of this typical CP/M machine with graphic capabilities is available (5), and discussion is restricted again only to those features of relevance. The basic components are a 5 inch screen providing a 24 line by 55 column window on the 32 lines by 128 column video memory, two floppy disk drives, a standard keyboard, and modem, IEEE488, and RS232 connectors. The Microsoft MBASIC language allowed access to all components, and custom-made external bridging circuits extended the external communication speed to a full range between 300 and 9600 bauds. Thus, the rates of the PU8610 interface of 300 and 4800 bauds could be matched, and connection between the two instruments was by means of a standard null modem or cross-over cable, as shown in Figure 1.

The next problem was to develop a software program to supervise the set-up, communication, sampling, and data handling routines in an automated fashion. The program should be menu driven, be flexible enough to handle different analytical requirements, and should generate printed or stored output in a compact form. The generalized sequence of operations is shown in Figure 2, and the host program written in MBASIC is listed in Appendix I. A detailed discussion of the major sections and subroutines of this program follows.

The first step is to set up the instruments and connections, and to load the program diskette on the Osborne. The graphic ingredients of the menus are contained in three assembler subroutines (Appendix II) which are directly stored in memory. The host program (Appendix I) first defines the PU8600 operating modes, the location of the assembler subroutines, and a series of special control characters (lines 460 to 820). It then enters the "choose mode" by setting up the main menu (Figure 3), and CONTROL-N (^N) steps through the available operating options (lines 1460 to 1960). The procedures may be terminated at this time by typing ^E which causes the computer to break the electronic link to the PU8610, clear the screen, and exit to CP/M (lines 2190 to 2210). If a particular operating mode is selected (^G) the program sets up a new menu (Figure 4) which lists the PU8610 parameter values (lines 2070 to 2150 and subroutine 4500). New values can be entered at the appropriate locations on the menu by means of ^R (lines 4500-5550).

The "collect data" mode is then scheduled (lines 2340 to 2880) based on the selected option. Initialization is effected by subroutine 5650 in which the computer transmits the codes to manipulate the PU8610 instrument settings. A command to load the sample and measure is given, and the computer memorizes and displays the returned results (lines 5880 to 5940). Subsequent samples are measured until the specified number of samples is reached, without changing the settings. For scanning purposes, however, the wavelength could be incremented through a simple modification of the program prior to remeasurement of a single sample.

The final step is a review of the collected data in lines 2990 to 4360 of the host program. The data are first retrieved from memory (lines 2990 to 3230) and the complete data set is displayed on the screen (Figure 5). The data set then can be printed (^H), saved on diskette (^S), and new data can

be measured in the same fashion (^N) or in a different mode (^E). Multiple samples of a single composition may have been specified (cycles >1), and the data sets contain the calculated averages only. Further manipulation of the data (regression analysis, component or baseline substractions, plots, etc.) also may be achieved with separate follow-up programs using the saved data files.

CONCLUSIONS

The PU8610 has proven to be an excellent replacement for the obsolete UV/VIS spectrophotometers in this laboratory. Interfacing with the Osborne extended the capabilities of this single beam instrument to match much more sophisticated and expensive models, while retaining the essential advantage of transportability. Measurements have shown that the analysis of copper and iron species in relatively clean solutions agrees with more elaborate titrations and AA spectrophotometric measurements. The PU8610 and the Osborne also have served as valuable learning tools for interfacing and data processing aspects, which should be further developed into complete laboratory automatization. If suitable accessories, instrumentation, and machine codes can be obtained, other potential applications include on-line analysis, and solution speciation in concentrated hydrometallurgical solutions. Thus, it should be possible to interface and synchronize automatic samplers and dilutors through the computer, but reliable translators for incompatible systems, zero dead volume switchable valves, short-path flow cells, and means to adjust the beam intensity and the optical slit width must be found.

REFERENCES

1. PU8600 Series UV/VIS Single Beam Spectrophotometers, User's Manual 9499 230 03911, Pye Unicam Ltd., Cambridge, England, 1984.
2. PU8600 Series Operating Guide, Pye Unicam Ltd., Cambridge, England, 1984.
3. Pye Unicam UV/VIS Applications, The PU8600 Series and Microcomputers, by P.J. Dowsing and V.M. Masters, 9498 904 92711, Pye Unicam Ltd., Cambridge, England.
4. Pye Unicam UV/VIS Single Beam Spectrophotometers, PU8600 & SP6 Series, Brochures and Fact Sheets, Pye Unicam Ltd., Cambridge, England.
5. Osborne 01 User's Reference Guide, by T. Hogan and M. Iannamico, Osborne Computer Corporation, Hayward, California, 1981.

Table 1 – Logical Key Codes for PU 8600 Series
ASCII key codes not included in the table are invalid.

LOGICAL KEY CODE (ASCII)	KEY INTERPRETATION	
+	Plus	
.	Decimal Point	
0	Digit 0	Digit Keys
1	Digit 1	
2	Digit 2	
3	Digit 3	
4	Digit 4	
5	Digit 5	
6	Digit 6	
7	Digit 7	
8	Digit 8	
9	Digit 9	
<	Remote Link ON	
>	Remote Link OFF	
A	ABS Mode	Mode Keys
B	Conc Stnd Mode	
C	Conc Factor Mode	
D	%T Mode	
F	Rate Mode	
G	Test Mode	
I	Wavelength Command	Command Keys
J	Peak Command	
K	Set Sample No.	
L	Store	
M	Recall	
N	List	
Q	°C	
R	Factor/Stnd Parameter	Parameter Keys
S	Delay Parameter	
T	Time Parameter	
U	Cycles Parameter	
V	★	
[	Print ADC ON	
]	Print ADC OFF	
a	Volume	Autocell Keys
b	Suckback	
c	Tube Calibration	
g	Fill Autocell	
h	Return	
i	1 Cell	Cell Programmer Keys
j	1,2 Cells	
k	1,2,3 Cells	
l	1,2,3,4 Cells	
m	1,2,3,4,5 Cells	
n	1,2,3,4,5,6 Cells	
p	Step	
w	Run/Stop	Front Panel Physical Keys
x	Enter	
y	D2 Lamp	
z	Zero	

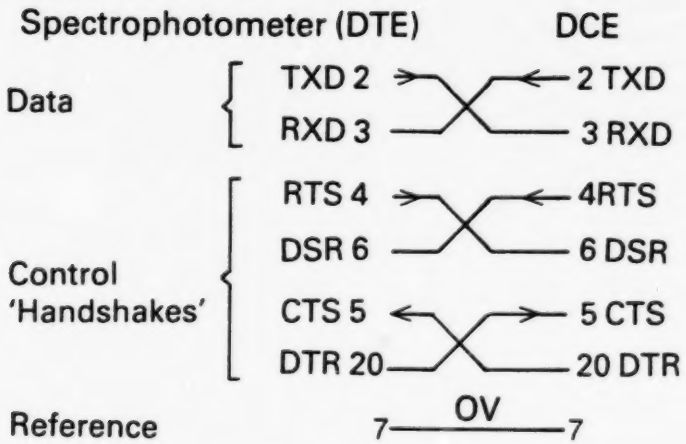

Figure 1 - RS232 Link between PU8610 and Osborne 01 via null modem

DTE = Data Terminal Equipment = PU8610

DCE = Data Communications Equipment = Osborne 01

TXD = Transmit Data

RXD = Receive Data

RTS = Ready To Send

0V = Signal Ground

DSR = Data Set Ready

CTS = Clear To Send

DTR = Data Terminal Ready

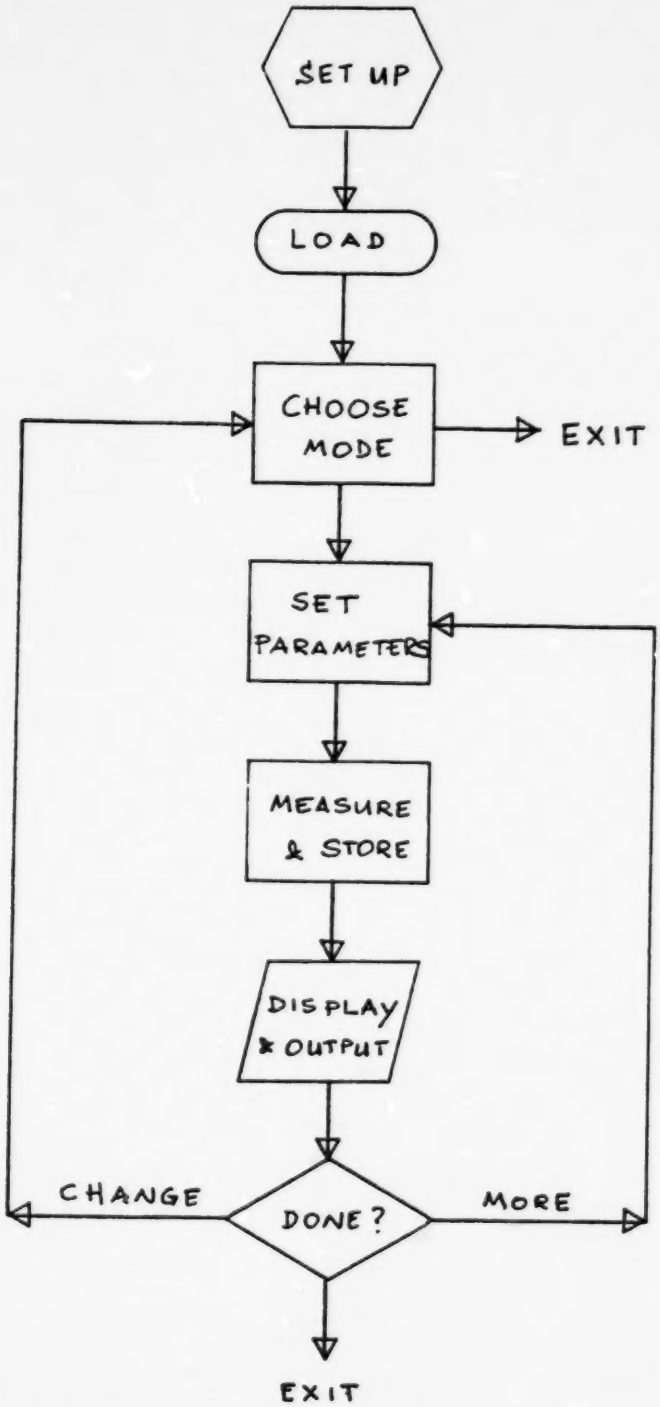

Figure 2 - Generalized Operational Flowsheet

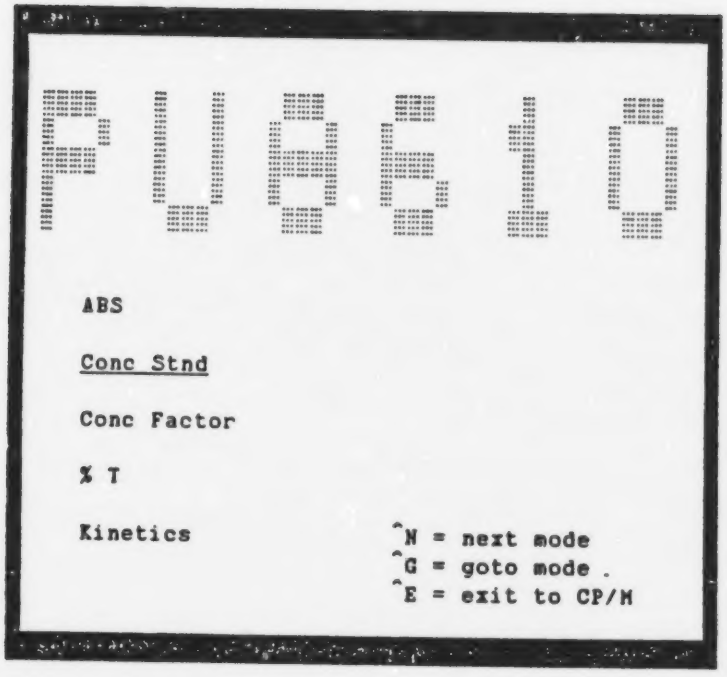

Figure 3 - 'Choose Mode' Menu

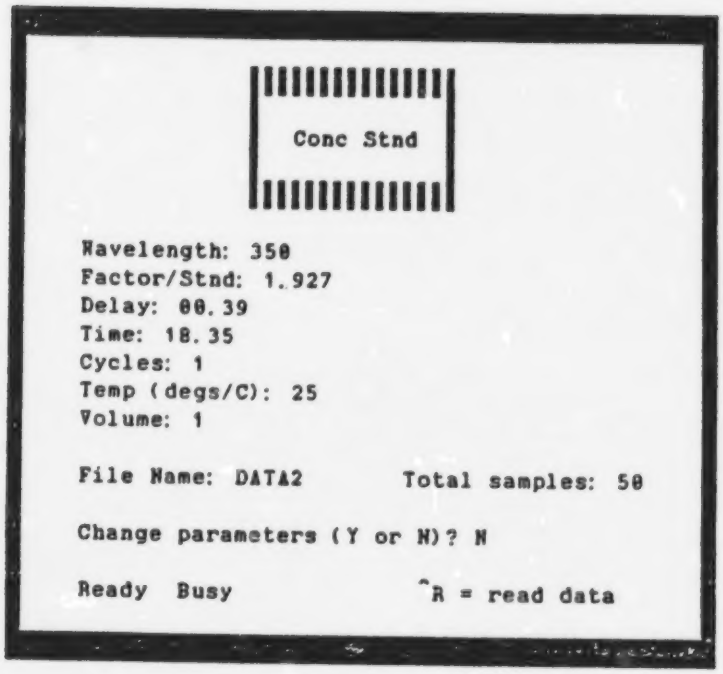

Figure 4 - Parameter Menu for Concentration Standard Mode

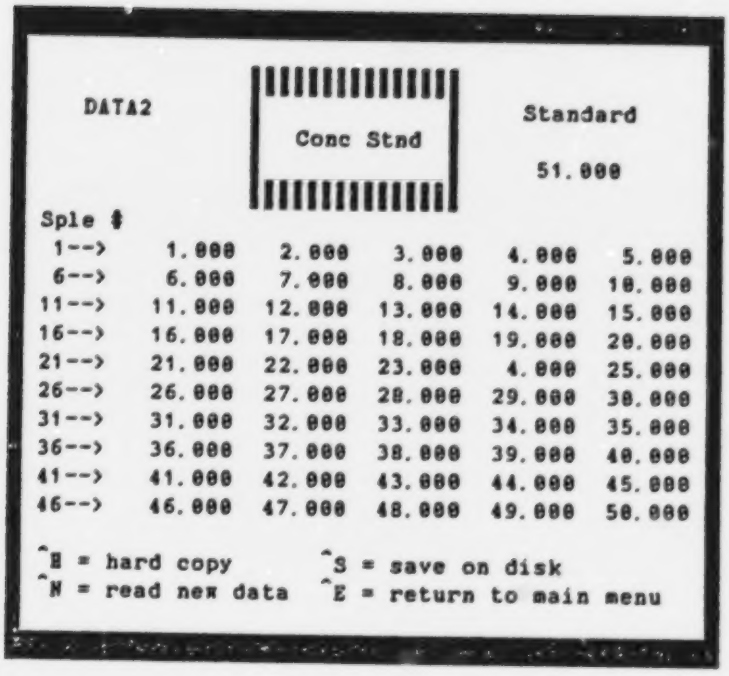

Figure 5 - Outlay of Data Output Format

APPENDIX I

```
10 REM
20 REM  ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
30 REM  ::                                                              ::
40 REM  :: PU8610BA is a BASIC program devised to link the              ::
50 REM  :: OSBORNE I microcomputer with the PYE UNICAM PU8610           ::
60 REM  :: spectrophotometer via their respective RS-232C in-          ::
70 REM  :: terface. Its main purpose is to make the microcom-           ::
80 REM  :: puter act as a virtual keyboard  replacing the               ::
90 REM  :: spectrophotometer's front keyboard. It allows the            ::
100 REM :: user to enter parameters, send commands, use an              ::
110 REM :: external printer instead of the spectrophotometer's          ::
120 REM :: internal one and to save the data collected on               ::
130 REM :: disk.                                                        ::
140 REM ::                                                              ::
150 REM :: PU8610BA is divided into three major sections:               ::
160 REM ::                                                              ::
170 REM ::                Choose mode                                   ::
180 REM ::                Collect data                                  ::
190 REM ::                Examine data                                  ::
200 REM ::                                                              ::
210 REM :: Choose mode: - point to one particular mode                  ::
220 REM ::              - choose one particular mode                    ::
230 REM ::              - break link with spectrophotometer,            ::
240 REM ::                exit PU8610BA and return to CP/M              ::
250 REM ::                system control                                ::
260 REM ::                                                              ::
270 REM :: Collect data: - enter parameters                             ::
280 REM ::               - link microcomputer with spectropho-          ::
290 REM ::                 tometer                                      ::
300 REM ::               - send parameters and commands to              ::
310 REM ::                 spectrophotometer                            ::
320 REM ::               - read data from spectrophotometer             ::
330 REM ::                                                              ::
340 REM :: Examine data: - display on screen data collected             ::
350 REM ::               - send a copy of the screen to a               ::
360 REM ::                 printer                                      ::
370 REM ::               - save data on disk                            ::
380 REM ::               - return to the Collect data section           ::
390 REM ::                 with the option of entering new              ::
400 REM ::                 parameters or retaining the ones             ::
410 REM ::                 previously entered                           ::
420 REM ::               - return to Choose mode                        ::
430 REM ::                                                              ::
440 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
450 REM
460 DATA "ABS","Conc Stnd","Conc Factor","% T","Kinetics"
470 DIM PU8610DATA(51,5)
480 REM
490 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
500 REM ::                                                              ::
510 REM :: Assembler subroutines used to upgrade the speed              ::
520 REM :: of execution of the graphics manipulations on the            ::
530 REM :: OSBORNE I screen.                                            ::
```

```
540 REM ::                                                          ::
550 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
560 REM
570 MAINFRAME=&HBE00
580 DRAWLOGO=&HBE86
590 TITLEFRAME=&HBE20
600 REM
610 REM ::::::::::::::::::::::::::::::::
620 REM ::                              ::
630 REM :: ASCII character definitions ::
640 REM ::                              ::
650 REM ::::::::::::::::::::::::::::::::
660 REM
670 NEWSCREEN$=CHR$(&H1A) :REM "CTRL Z"
680 ESC$=CHR$(&H1B) :REM "esc"
690 UON$=CHR$(&H6C) :REM "l"
700 UOFF$=CHR$(&H6D) :REM "m"
710 HON$=CHR$(&H29) :REM ")"
720 FON$=CHR$(&H28) :REM "("
730 BELL$=CHR$(&H7) :REM "BELL"
740 EQUAL$=CHR$(&H3D) :REM "="
750 CTRLN$=CHR$(&HE) :REM "CTRL N"
760 CTRLG$=CHR$(&H7) :REM "CTRL G"
770 CTRLE$=CHR$(&H5) :REM "CTRL E"
780 CTRLR$=CHR$(&H12) :REM "CTRL R"
790 CTRLS$=CHR$(&H13) :REM "CTRL S"
800 CTRLH$=CHR$(&H8) :REM "CTRL H"
810 ASTERISK$=CHR$(&H2A) :REM "*"
820 FORMFEED$=CHR$(&HC) :REM "CTRL L"
830 REM
840 REM  ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
850 REM  ::                                                          ::
860 REM  :: OSBORNE I direct screen manipulations using              ::
870 REM  :: special ASCII character sequences.                       ::
880 REM  ::                                                          ::
890 REM  :: a) PRINT ESC$;HON$;                                      ::
900 REM  ::                                                          ::
910 REM  ::    starts the half intensity display mode.               ::
920 REM  ::                                                          ::
930 REM  :: b) PRINT ESC$;FON$;                                      ::
940 REM  ::                                                          ::
950 REM  ::    starts the full intensity display mode.               ::
960 REM  ::                                                          ::
970 REM  :: c) PRINT ESC$;UON$;                                      ::
980 REM  ::                                                          ::
990 REM  ::    starts underlining all characters.                    ::
1000 REM ::                                                          ::
1010 REM :: d) PRINT ESC$;UOFF$;                                     ::
1020 REM ::                                                          ::
1030 REM ::    ends underlining all characters.                      ::
1040 REM ::                                                          ::
1050 REM :: e) PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);        ::
1060 REM ::                                                          ::
1070 REM ::    positions the cursor on the line specified by         ::
1080 REM ::    YPOS and the column specified by XPOS referen-        ::
```

```
1090 REM ::     ced  to the upperleft corner of the screen.     ::
1100 REM ::                                                     ::
1110 REM ::     ex: YPOS=6 : XPOS=21                            ::
1120 REM ::                                                     ::
1130 REM ::     would position the cursor on line 6, column 21  ::
1140 REM ::                                                     ::
1150 REM :: f) PRINT NEWSCREEN$;                                ::
1160 REM ::                                                     ::
1170 REM ::     clears the screen and homes the cursor.         ::
1180 REM ::     YPOS=1 : XPOS=1                                 ::
1190 REM ::                                                     ::
1200 REM :: g) PRINT BELL$;                                     ::
1210 REM ::                                                     ::
1220 REM ::     sounds the console bell.                        ::
1230 REM ::                                                     ::
1240 REM :: Printer ASCII sequences:                            ::
1250 REM ::                                                     ::
1260 REM :: a) PRINT ASTERISK$;                                 ::
1270 REM ::                                                     ::
1280 REM ::     sends the ASCII character "*" to the printer.   ::
1290 REM ::                                                     ::
1300 REM :: b) PRINT FORMFEED$;                                 ::
1310 REM ::                                                     ::
1320 REM ::     sends a form feed command to the printer.       ::
1330 REM ::                                                     ::
1340 REM :::::::::::::::::::::::::::::::::::::::::::::::::::::::::
1350 REM
1360 REM
1370 REM
1380 REM ::::::::::::::::::::::::::
1390 REM ::                        ::
1400 REM ::     Choose mode        ::
1410 REM ::                        ::
1420 REM ::::::::::::::::::::::::::
1430 REM
1440 REM ------ Set up screen ------
1450 REM
1460 CALL MAINFRAME
1470 CALL DRAWLOGO
1480 PRINT ESC$;HON$;
1490 YPOS=19:XPOS=28
1500 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
1510 PRINT "^N = next mode"
1520 YPOS=20:XPOS=28
1530 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
1540 PRINT "^G = goto mode"
1550 YPOS=21:XPOS=28
1560 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
1570 PRINT "^E = exit to CP/M"
1580 PRINT ESC$;HON$;
1590 RESTORE
1600 YLINE=11
1610 FOR I=1 TO 5
1620 YPOS=YLINE:XPOS=6
1630 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
```

```
1640 READ MODE$
1650 PRINT MODE$
1660 YLINE=YLINE+2
1670 NEXT I
1680 REM
1690 REM ----- Point to mode -----
1700 REM
1710 RESTORE
1720 YLINE=11
1730 FOR I=1 TO 5
1740 PRINT ESC$;FON$;
1750 YPOS=YLINE:XPOS=6
1760 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
1770 READ MODE$
1780 PRINT ESC$;UON$;MODE$
1790 REM
1800 REM ----- Enter command (CTRL N, CTRL G, CTRL E) -----
1810 REM
1820 A$=INPUT$(1)
1830 IF A$=CTRLN$ THEN 1910
1840 IF A$=CTRLG$ THEN 2010
1850 IF A$=CTRLE$ THEN 2190
1860 PRINT BELL$
1870 GOTO 1820
1880 REM
1890 REM ----- CTRL N = point to next mode
1900 REM
1910 YPOS=YLINE:XPOS=6
1920 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
1930 PRINT ESC$;HON$;
1940 PRINT ESC$;UOFF$;MODE$
1950 YLINE=YLINE+2
1960 NEXT I
1970 REM
1980 REM ----- CTRL G = mode selected -----
1990 REM
2000 GOTO 1710
2010 GOSUB 6890
2020 IF MODE$="ABS" THEN XCOL=24:GOTO 2070
2030 IF MODE$="Conc Stnd" THEN XCOL=21:GOTO 2070
2040 IF MODE$="Conc Factor" THEN XCOL=20:GOTO 2070
2050 IF MODE$="% T" THEN XCOL=24:GOTO 2070
2060 XCOL=21
2070 PRINT ESC$;UOFF$
2080 CALL MAINFRAME
2090 CALL TITLEFRAME
2100 YPOS=5:XPOS=XCOL
2110 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
2120 PRINT MODE$
2130 GOSUB 4500
2140 IF MODE$="ABS" OR MODE$="% T" THEN 2600
2150 GOTO 2340
2160 REM
2170 REM ----- CTRL E = exit to CP/M -----
2180 REM
```

```
2190 LPRINT ">" :REM break link with spectrophotometer
2200 PRINT NEWSCREEN$
2210 SYSTEM
2220 REM
2230 REM ::::::::::::::::::::::::::
2240 REM ::                        ::
2250 REM ::     Collect data       ::
2260 REM ::                        ::
2270 REM ::::::::::::::::::::::::::
2280 REM
2290 REM
2300 REM              ----- Read standard -----
2310 REM
2320 REM    Conc Stnd, Conc Factor, Kinetics modes only
2330 REM
2340 IF SAMESTANDARD$="N" THEN 2600
2350 YPOS=19:XPOS=6
2360 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
2370 PRINT "Sample: Standard"
2380 COLLECT=0
2390 FOR SAMPLE=1 TO SAMPLESET
2400 GOSUB 6300
2410 YPOS=20:XPOS=15
2420 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
2430 PRINT SAMPLE
2440 REM
2450 REM ----- CTRL R = read standard -----
2460 REM
2470 A$=INPUT$(1)
2480 IF A$=CTRLR$ THEN 2510
2490 PRINT BELL$
2500 GOTO 2470
2510 GOSUB 6130
2520 IF DATASET$="NEW" AND SAMPLE=1 THEN 2550
2530 GOSUB 5880
2540 GOTO 2560
2550 GOSUB 5650
2560 NEXT SAMPLE
2570 REM
2580 REM ----- Read samples -----
2590 REM
2600 FOR COLLECT=1 TO TOTALSAMPLES
2610 YPOS=19:XPOS=6
2620 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
2630 PRINT "Sample:          "
2640 YPOS=19:XPOS=14
2650 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
2660 PRINT COLLECT
2670 FOR SAMPLE=1 TO SAMPLESET
2680 GOSUB 6300
2690 YPOS=20:XPOS=15
2700 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
2710 PRINT SAMPLE
2720 REM
2730 REM ----- CTRL R = read one set of data -----
```

```
2740 REM
2750 A$=INPUT$(1)
2760 IF A$=CTRLR$ THEN 2790
2770 PRINT BELL$
2780 GOTO 2750
2790 GOSUB 6130
2800 IF MODE$="ABS" OR MODE$="% T" THEN 2830
2810 GOSUB 5880
2820 GOTO 2870
2830 IF DATASET$="NEW" THEN 2860
2840 GOSUB 5880
2850 GOTO 2870
2860 GOSUB 5650
2870 NEXT SAMPLE
2880 NEXT COLLECT
2890 REM
2900 REM ::::::::::::::::::::::::
2910 REM ::                      ::
2920 REM ::    Examine data      ::
2930 REM ::                      ::
2940 REM ::::::::::::::::::::::::
2950 REM
2960 REM
2970 REM ----- Set up screen -----
2980 REM
2990 YPOS=8:XPOS=3
3000 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3010 PRINT "Sple #"
3020 YLINE=9
3030 FOR I=1 TO 46 STEP 5
3040 YPOS=YLINE:XPOS=3
3050 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3060 PRINT USING "##";I;
3070 PRINT "--> ";SPC(39)
3080 PRINT
3090 YLINE=YLINE+1
3100 NEXT I
3110 YPOS=19:XPOS=6
3120 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3130 PRINT SPC(20)
3140 PRINT
3150 PRINT ESC$;HON$;
3160 YPOS=20:XPOS=3
3170 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3180 PRINT "^H = hard copy";SPC(6)"^S = save on disk"
3190 YPOS=21:XPOS=3
3200 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3210 PRINT "^N = read new data";SPC(2)"^E = return to choose mode"
3220 YPOS=22:XPOS=3
3230 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3240 REM
3250 REM ----- Print filename -----
3260 REM
3270 PRINT SPC(14)
3280 PRINT ESC$;FON$;
```

```
3290 YPOS=4:XPOS=6
3300 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3310 FILENAME$=LEFT$(FILENAME$,8)
3320 PRINT FILENAME$;
3330 IF MODE$="ABS" OR MODE$="% T" THEN 3510
3340 REM
3350 REM ----- Print standard -----
3360 REM
3370 YPOS=4:XPOS=36
3380 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3390 PRINT "Standard"
3400 YPOS=6:XPOS=36
3410 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3420 COLLECT=0
3430 AVERAGE=0
3440 FOR SAMPLE=1 TO SAMPLESET
3450 AVERAGE=AVERAGE+PU8610DATA(COLLECT,SAMPLE)
3460 NEXT SAMPLE
3470 PRINT USING "###.###";AVERAGE/SAMPLESET
3480 REM
3490 REM ----- Print samples -----
3500 REM
3510 YLINE=9
3520 FOR COLLECT=1 TO TOTALSAMPLES STEP 5
3530 YPOS=YLINE:XPOS=9
3540 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3550 FOR PRINTAVERAGE=COLLECT TO COLLECT+4
3560 IF PRINTAVERAGE>TOTALSAMPLES THEN 3670
3570 AVERAGE=0
3580 FOR SAMPLE=1 TO SAMPLESET
3590 AVERAGE=AVERAGE+PU8610DATA(PRINTAVERAGE,SAMPLE)
3600 NEXT SAMPLE
3610 PRINT USING "###.###";AVERAGE/SAMPLESET;
3620 PRINT " ";
3630 NEXT PRINTAVERAGE
3640 PRINT
3650 YLINE=YLINE+1
3660 NEXT COLLECT
3670 FILESAVED$="NO"
3680 YPOS=22:XPOS=1
3690 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3700 REM
3710 REM ----- Enter command (CTRL N, CTRL H,CTRL E, CTRL S) -----
3720 REM
3730 A$=INPUT$(1)
3740 IF A$=CTRLN$ THEN 3980
3750 IF A$=CTRLH$ THEN 4120
3760 IF A$=CTRLE$ THEN 4050
3770 IF A$=CTRLS$ THEN 3830
3780 PRINT BELL$
3790 GOTO 3730
3800 REM
3810 REM ----- CTRL S = save data on disk under filename -----
3820 REM
3830 GOSUB 6480
```

```
3840 YPOS=20:XPOS=23
3850 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3860 PRINT ESC$;HON$;
3870 PRINT "^S = save on disk"
3880 PRINT ESC$;FON$;
3890 YPOS=22:XPOS=3
3900 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
3910 FILESAVED$="YES"
3920 IF A$=CTRLN$ THEN 2020
3930 IF A$=CTRLE$ THEN 1460
3940 GOTO 3730
3950 REM
3960 REM ----- CTRL N = collect more data using same mode -----
3970 REM
3980 IF FILESAVED$="YES" THEN 2020
3990 GOSUB 6720
4000 IF SAVEFILE$="N" THEN 2020
4010 GOTO 3830
4020 REM
4030 REM ----- CTRL E = return to Choose mode -----
4040 REM
4050 IF FILESAVED$="YES" THEN 1460
4060 GOSUB 6720
4070 IF SAVEFILE$="N" THEN 1460
4080 GOTO 3830
4090 REM
4100 REM ----- CTRL H = send copy of screen to external printer -----
4110 REM
4120 YPOS=20:XPOS=3
4130 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4140 PRINT "^H = hard copy";
4150 FOR I=&HF000 TO &HF900 STEP 128
4160 FOR J=I TO I+49
4170 B=PEEK(J)
4180 IF B=150 OR B=22 OR B=1 THEN 4210
4190 LPRINT CHR$(B);
4200 GOTO 4220
4210 LPRINT ASTERISK$;
4220 NEXT J
4230 LPRINT
4240 NEXT I
4250 FOR I=1 TO 50
4260 LPRINT ASTERISK$;
4270 NEXT I
4280 LPRINT FORMFEED$
4290 YPOS=20:XPOS=3
4300 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4310 PRINT ESC$;HON$;
4320 PRINT "^H = hard copy"
4330 PRINT ESC$;FON$;
4340 YPOS=22:XPOS=3
4350 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4360 GOTO 3730
4370 REM
4380 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
```

```
4390 REM ::                                                    ::
4400 REM :: Subroutine to enter all the parameters neces-      ::
4410 REM :: sary for a successfull run                         ::
4420 REM ::                                                    ::
4430 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
4440 REM
4450 REM
4460 REM ----- Print the parameters along with their values -----
4470 REM
4480 REM       note: if new run then parameter values will be blank
4490 REM
4500 YPOS=9:XPOS=6
4510 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4520 PRINT "Wavelength: ";WL$
4530 YPOS=10:XPOS=6
4540 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4550 PRINT "Factor/Stnd: ";FAC$
4560 YPOS=11:XPOS=6
4570 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4580 PRINT "Delay: ";DEL$
4590 YPOS=12:XPOS=6
4600 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4610 PRINT "Time: ";TIM$
4620 YPOS=13:XPOS=6
4630 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4640 PRINT "Cycles: ";CYC$
4650 YPOS=14:XPOS=6
4660 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4670 PRINT "Temp (degs/C): ";TEM$
4680 YPOS=15:XPOS=6
4690 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4700 PRINT "Volume: ";VOL$
4710 YPOS=17:XPOS=6
4720 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4730 PRINT "File Name:";SPC(13)"Total samples:    "
4740 YPOS=18:XPOS=6
4750 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4760 PRINT "Readings/sample:     "
4770 YPOS=19:XPOS=6
4780 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4790 PRINT "Sample:";SPC(8)
4800 YPOS=20:XPOS=6
4810 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4820 PRINT "Reading:"
4830 GOSUB 6130
4840 PRINT ESC$;HON$;
4850 YPOS=21:XPOS=28
4860 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4870 PRINT "^R = read data"
4880 PRINT ESC$;FON$;
4890 IF DATASET$="SAME" THEN 5140
4900 REM
4910 REM ----- Enter new parameter values -----
4920 REM
4930 YPOS=9:XPOS=17
```

```
4940 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4950 INPUT " ",WL$
4960 YPOS=10:XPOS=18
4970 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
4980 INPUT " ",FAC$
4990 YPOS=11:XPOS=12
5000 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5010 INPUT " ",DEL$
5020 YPOS=12:XPOS=11
5030 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5040 INPUT " ",TIM$
5050 YPOS=13:XPOS=13
5060 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5070 INPUT " ",CYC$
5080 YPOS=14:XPOS=20
5090 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5100 INPUT " ",TEM$
5110 YPOS=15:XPOS=13
5120 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5130 INPUT " ",VOL$
5140 YPOS=17:XPOS=17
5150 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5160 INPUT;"",FILENAME$
5170 YPOS=17:XPOS=44
5180 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5190 INPUT "",TOTALSAMPLES
5200 IF TOTALSAMPLES<=50 THEN 5260
5210 YPOS=17:XPOS=44
5220 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5230 PRINT SPC(3)
5240 PRINT BELL$
5250 GOTO 5170
5260 YPOS=18:XPOS=23
5270 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5280 INPUT "",SAMPLESET
5290 IF SAMPLESET<=5 THEN 5350
5300 YPOS=18:XPOS=23
5310 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5320 PRINT SPC(2)
5330 PRINT BELL$
5340 GOTO 5260
5350 YPOS=22:XPOS=6
5360 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5370 INPUT;"Change parameters (Y or N)";CHANGE$
5380 YPOS=22:XPOS=6
5390 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5400 PRINT SPC(29)
5410 PRINT
5420 IF CHANGE$="N" THEN 5450
5430 GOSUB 6890
5440 GOTO 4500
5450 YPOS=22:XPOS=6
5460 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5470 IF DATASET$="NEW" OR MODE$="ABS" OR MODE$="% T" THEN 5550
5480 INPUT;"Read new standard (Y or N)";SAMESTANDARD$
```

```
5490 YPOS=22:XPOS=6
5500 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5510 PRINT SPC(29)
5520 PRINT
5530 YPOS=19:XPOS=6
5540 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
5550 RETURN
5560 REM
5570 REM ::::::::::::::::::::::::::::::::::::::::::::::::
5580 REM ::                                              ::
5590 REM :: Subroutine to send the logical key codes     ::
5600 REM :: to the spectrophotometer representing        ::
5610 REM :: the parameters and read data command         ::
5620 REM ::                                              ::
5630 REM ::::::::::::::::::::::::::::::::::::::::::::::::
5640 REM
5650 LPRINT "<"; :REM set up link
5660 GOSUB 6020
5670 LPRINT "I";WL$;"x"; :REM Wavelength;enter;
5680 GOSUB 6020
5690 LPRINT "R";FAC$;"x"; :REM Factor;enter;
5700 GOSUB 6020
5710 LPRINT "S";DEL$;"x"; :REM Delay;enter;
5720 GOSUB 6020
5730 LPRINT "T";TIM$;"x"; :REM Time;enter;
5740 GOSUB 6020
5750 LPRINT "U";CYC$;"x"; :REM Cycle;enter;
5760 GOSUB 6020
5770 LPRINT "Q";TEM$;"x"; :REM Temperature;enter
5780 GOSUB 6020
5790 LPRINT "a";VOL$;"x"; :REM Volume;enter
5800 GOSUB 6020
5810 IF MODE$="ABS" THEN LPRINT "Ax":GOTO 5860
5820 IF MODE$="Conc Stnd" THEN LPRINT "Bx":GOTO 5860
5830 IF MODE$="Conc Factor" THEN LPRINT "Cx":GOTO 5860
5840 IF MODE$="% T" THEN LPRINT "Dx":GOTO 5860
5850 LPRINT "Fx"
5860 DATASET$="SAME"
5870 GOSUB 6020
5880 LPRINT "g"; :REM Fill autocell
5890 POKE 3,41 :REM change CP/M IOBYTE FROM TTY to CRT
5900 LPRINT
5910 INPUT PU8610DATA$
5920 PU8610DATA(COLLECT,SAMPLE)=VAL(RIGHT$(PU8610DATA$,7))
5930 POKE 3,&H40 :REM change CP/M IOBYTE FROM CRT to TTY
5940 RETURN
5950 REM
5960 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
5970 REM ::                                                    ::
5980 REM :: Subroutine generating a delay of appr. 3 seconds   ::
5990 REM ::                                                    ::
6000 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
6010 REM
6020 FOR I=1 TO 1000
6030 NEXT I
```

```
6040 RETURN
6050 REM
6060 REM ::::::::::::::::::::::::::::::::::::::::::::::::::
6070 REM ::                                                ::
6080 REM :: Subroutine to print "Ready" half intensity     ::
6090 REM :: and "Busy" full intensity                      ::
6100 REM ::                                                ::
6110 REM ::::::::::::::::::::::::::::::::::::::::::::::::::
6120 REM
6130 YPOS=22:XPOS=6
6140 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6150 PRINT ESC$;HON$;
6160 PRINT "Ready  ";
6170 PRINT ESC$;FON$;
6180 PRINT "Busy     "
6190 YPOS=22:XPOS=1
6200 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6210 RETURN
6220 REM
6230 REM ::::::::::::::::::::::::::::::::::::::::::::::::::
6240 REM ::                                                ::
6250 REM :: Subroutine to print "Ready" full intensity     ::
6260 REM :: and "Busy" half intensity                      ::
6270 REM ::                                                ::
6280 REM ::::::::::::::::::::::::::::::::::::::::::::::::::
6290 REM
6300 YPOS=22:XPOS=6
6310 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6320 PRINT ESC$;FON$;
6330 PRINT "Ready  ";
6340 PRINT ESC$;HON$;
6350 PRINT "Busy          "
6360 PRINT ESC$;FON$;
6370 YPOS=23:XPOS=1
6380 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6390 RETURN
6400 REM
6410 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
6420 REM ::                                                    ::
6430 REM :: Subroutine to save data on disk under filename     ::
6440 REM :: entered previously                                 ::
6450 REM ::                                                    ::
6460 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::
6470 REM
6480 PRINT
6490 YPOS=20:XPOS=23
6500 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6510 PRINT ESC$;FON$;
6520 PRINT "^S = save on disk";
6530 OPEN "O",#1,FILENAME$
6540 FOR I=0 TO TOTALSAMPLES
6550 AVERAGE=0
6560 FOR J=1 TO SAMPLESET
6570 AVERAGE=AVERAGE+PU8610DATA(I,J)
6580 NEXT J
```

```
6590 AVERAGE=AVERAGE/SAMPLESET
6600 WRITE #1,AVERAGE
6610 NEXT I
6620 CLOSE #1
6630 RETURN
6640 REM
6650 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
6660 REM ::                                                              ::
6670 REM :: Subroutine determining whether data is to be                 ::
6680 REM :: saved or not                                                 ::
6690 REM ::                                                              ::
6700 REM ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
6710 REM
6720 YPOS=22:XPOS=3
6730 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6740 PRINT "Save ";FILENAME$;" (Y or N)";
6750 INPUT;SAVEFILE$
6760 IF SAVEFILE$="N" THEN RETURN
6770 PRINT
6780 YPOS=22:XPOS=3
6790 PRINT ESC$;EQUAL$;CHR$(YPOS+31);CHR$(XPOS+31);
6800 PRINT SPC(30)
6810 RETURN
6820 REM
6830 REM ::::::::::::::::::::::::::::::::::::
6840 REM ::                                ::
6850 REM :: Subroutine to reset parameters ::
6860 REM ::                                ::
6870 REM ::::::::::::::::::::::::::::::::::::
6880 REM
6890 DATASET$="NEW"
6900 SAMESTANDARD$="Y"
6910 WL$="          "
6920 FAC$="          "
6930 DEL$="          "
6940 TIM$="          "
6950 CYC$="          "
6960 TEM$="          "
6970 VOL$="          "
6980 RETURN
```

SYMBOL	REFERENCE LINE								
1460	3930	4050	4070						
1710	2000								
1820	1870								
1910	1830								
2010	1840								
2020	3920	3980	4000						
2070	2020	2030	2040	2050					
2190	1850								
2340	2150								
2470	2500								
2510	2480								
2560	2540								
2600	2140	2340							
2750	2780								
2790	0								
2830	2800								
2870	2820	2850							
3510	3330								
3670	3560								
3730	3790	3940	4360						
3830	3770	4010	4080						
3980	3740								
4050	3760								
4120	3750								
4210	4180								
4220	4200								
4500	2130	5440							
5170	5250								
5260	5200	5340							
5350	5290								
5450	5420								
5650	2550	2860							
5860	5810	5820	5830	5840					
5880	2530	2810	2840						
6020	5660	5680	5700	5720	5740	5760	5780	5800	5870
6130	2510	2790	4830						
6300	2400	2680							
6480	3830								
6720	3990	4060							

SYMBOL	REFERENCE LINE										
6890	2010	5430									
A$	1820	1830	1840	1850	2470	2480	2750	0	3730	3740	3750
	3760	3770	3920	3930							
AVED$	3670	3910	3980	4050							
AVERAGE	3430	3450	3470	3550	3560	3570	3590	3610	3630	6550	6570
	6590	6600									
B	4170	4180	4190								
BELL$	730	1860	2490	2770	3780	5240	5330				
CHANGE$	5370	5420									
COLLECT	2380	2600	2660	2880	3420	3450	3520	3550	3660	5920	
CIRLE$	770	1850	3760	3930							
CIRLG$	760	1840									
CIRLH$	800	3750									
CIRLN$	750	1830	3740	3920							
CIRLR$	780	2480	0								
CIRLS$	790	3770									
CYC$	4640	5070	5750	6950							
DEL$	4580	5010	5710	6930							
DRAWLOOD	580	1470									
E$	1640	1650	1770	1780	1940	2020	2030	2040	2050	2120	2140
	2800	3330	5810	5820	5830	5840					
EQUAL$	740	1500	1530	1560	1630	1760	1920	2110	2360	2420	2620
	2650	2700	3000	3050	3120	3170	3200	3230	3300	3380	3410
	3540	3690	3850	3900	4130	4300	4350	4510	4540	4570	4600
	4630	4660	4690	4720	4750	4780	4810	4860	4940	4970	5000
	5030	5060	5090	5120	5150	5180	5220	5270	5310	5360	5390
	5460	5500	5540	6140	6200	6310	6380	6500	6730	6790	
ESC$	680	1480	1500	1530	1560	1580	1630	1740	1760	1780	1920
	1930	1940	2070	2110	2360	2420	2620	2650	2700	3000	3050
	3120	3150	3170	3200	3230	3280	3300	3380	3410	3540	3690
	3850	0	3880	3900	4130	4300	4310	4330	4350	4510	4540
	4570	4600	4630	4660	4690	4720	4750	4780	4810	4840	4860
	4880	4940	4970	5000	5030	5060	5090	5120	5150	5180	5220
	5270	5310	5360	5390	5460	5500	5540	6140	6150	6170	6200
	6310	6320	6340	6360	6380	6500	6510	6730	6790		
FAC$	4550	4980	5690	6920							
FILE$	4000	4070	6750	6760							
FILENAME$	0	3320	5160	6530	6740						
FCN$	720	1740	3280	3880	4330	4880	6170	6320	6360	6510	
HCN$	710	1480	1580	1930	3150	0	4310	4840	6150	6340	
I	1610	1670	1730	1960	3030	3060	3100	4150	4160	4240	4250
	4270	6020	6030	6540	6570	0					
J	4160	4170	4220	6560	6570	6580					

SYMBOL	REFERENCE LINE										
MAINFRAME	570	1460	2080								
MFEED$	820	4280									
PUB610DATA$	5910	5920									
PUB610DATA(	470	3450	3590	5920	6570						
SAMESTANDARD$	2340	5480	6900								
SAMPLE	2390	2430	2560	2670	2710	2870	3440	3450	3460	3580	3590
	3600	5920									
SAMPLESET	2390	2670	3440	3470	3580	3610	5280	5290	6560	6590	
SCREEN$	670	2200									
TALSAMPLES	2600	3520	3560	5190	5200	6540					
TEM$	4670	5100	5770	6960							
TERISK$	810	4210	4260								
TIM$	4610	5040	5730	6940							
TITLEFRAME	590	2090									
UOFF$	700	1940	2070								
UON$	690	1780									
VOL$	4700	5130	5790	6970							
WL$	4520	4950	5670	6910							
XCOL	2020	2030	2040	2050	2060	2100					
XPOS	1490	1500	1520	1530	1550	1560	1620	1630	1750	1760	1910
	1920	2100	2110	2350	2360	2410	2420	2610	2620	2640	2650
	2690	2700	2990	3000	3040	3050	3110	3120	3160	3170	3190
	3200	3220	3230	3290	3300	3370	3380	3400	3410	3530	3540
	3680	3690	3840	3850	3890	3900	4120	4130	4290	4300	4340
	4350	4500	4510	4530	4540	4560	4570	4590	4600	4620	4630
	4650	4660	4680	4690	4710	4720	4740	4750	4770	4780	4800
	4810	4850	4860	4930	4940	0	4970	4990	5000	5020	5030
	5050	5060	5080	5090	5110	5120	5140	5150	5170	5180	5210
	5220	5260	5270	5300	5310	5350	5360	5380	5390	5450	5460
	5490	5500	5530	5540	6130	6140	6190	6200	6300	6310	6370
	6380	6490	6500	6720	6730	6780	6790				
YLINE	1600	1620	0	1720	1750	1910	1950	3020	3040	3090	3510
	3530	3650									
YPOS	1490	1500	1520	1530	1550	1560	1620	1630	1750	1760	1910
	1920	2100	2110	2350	2360	2410	2420	2610	2620	2640	2650
	2690	2700	2990	3000	3040	3050	3110	3120	3160	3170	3190
	3200	3220	3230	3290	3300	3370	3380	3400	3410	3530	3540
	3680	3690	3840	3850	3890	3900	4120	4130	4290	4300	4340
	4350	4500	4510	4530	4540	4560	4570	4590	4600	4620	4630
	4650	4660	4680	4690	4710	4720	4740	4750	4770	4780	4800
	4810	4850	4860	4930	4940	0	4970	4990	5000	5020	5030
	5050	5060	5080	5090	5110	5120	5140	5150	5170	5180	5210
	5220	5260	5270	5300	5310	5350	5360	5380	5390	5450	5460
	5490	5500	5530	5540	6130	6140	6190	6200	6300	6310	6370
	6380	6490	6500	6720	6730	6780	6790				

LINES: 698 BYTES: 23456 SYMBOLS: 87 REFERENCES: 726

APPENDIX II

```
;
; ####  #   #  ###   ###    #    ###    #    ###
; #   # #   # #   # #      ##   #   #  # #  #
; ####  #   #  ###  ####    #   #   # ##### ###
; #     #  # #   # #   #    #   #  ## #   #    #
; #      ###   ###   ###   ###   ### #   #  ###
;
;
;               Version: 1.01
;               Date: April 21, 1986
;               Author: Pierre Richer
;
;
; PU8610AS is a group of three subroutines called by the host
; program PU8610BA. Its main purpose is to speed up the graphics
; manipulations on the OSBORNE I screen. The three subroutines are:
;
;               MainFrame
;               TitleFrame
;               DrawLogo

;
; CP/M BDOS parameters
;

0005 =          Bdos EQU 5 ; entry vector for BDOS functions
0002 =          PrintCharacter EQU 2 ; BDOS function 2

;
; Cursor movement hex codes for the OSBORNE I screen
;

001A =          ClearHome EQU 1AH ; clears the screen & homes the cursor.

;
; Miscellaneous hex codes for the OSBORNE I screen
;

001B =          Esc EQU 1BH ; escape key.
0067 =          Gon EQU 67H ; starts graphics character interpretation.
0047 =          Goff EQU 47H ; ends graphics character interpretation.
0020 =          Space EQU 20H ; space bar.
B000            ORG 0B000H

;
; MainFrame...front end routine for the subroutine DrawFrame.
;
;               It supplies the origin, the width, the height
;               and the graphics character to be used for drawing
;               the rectangle (frame) enclosing all of the various
;               displays occuring during the operation of the
;               host program PU8610BA.
;
```

```
                        MainFrame:

B000 CD0DB1              CALL NewScreen ; clear screen, home cursor.
B003 2100F0              LXI H,0F000H ; origin of main frame.
B006 22B4B1              SHLD X0Y0
B009 3E32                MVI A,50 ; width = 50 characters.
B00B 32B6B13E15          STA Width
B00E 3E15                MVI A,21 ; height = 21 lines.
B010 32B7B1              STA Height
B013 3E16                MVI A,16H ; graphics character for frame.
B015 32B8B1              STA GraphicsChr
B018 CD00B1              CALL GraphicsOn ; enable graphics on the screen.
B01B C336B0              JMP DrawFrame

          ;
          ; TitleFrame...front end routine for the subroutine DrawFrame.
          ;
          ;               It supplies the origin, the width, the height and
          ;               the graphics character to draw the frame enclo-
          ;               sing the mode chosen by the user.
          ;

                        TitleFrame:

B01E 2111F1              LXI H,0F111H ; origin of title frame.
B021 22B4B1              SHLD X0Y0
B024 3E0F                MVI A,15 ; width = 15 characters.
B026 32B6B1                   STA Width
B029 3E03                MVI A,3 ; height = 3 lines.
B02B 32B7B1              STA Height
B02E 3E01                MVI A,1 ; graphics character for the titleframe.
B030 32B8B1              STA GraphicsChr
B033 CD00B1              CALL GraphicsOn ; enable graphics on the screen.

          ;
          ; DrawFrame...subroutine to draw a rectangle using the para-
          ;             meters supplied by previous front end routines.
          ;
          ;
          ;
          ;       (X0,Y0)_________(Xx,Y0)_________(Xwidth,Y0)
          ;       (X0,Y1)|                        |(Xwidth,Y1)
          ;              |                        |
          ;              |                        |
          ;              |                        |
          ;              |                        |
          ;      (X0,Yy1)|         (Xx,Yy)        |(Xx,Yy2)
          ;              |                        |
          ;              |                        |
          ;              |                        |
          ;              |                        |
          ;(X0,Yheight)  |                        |(Xwidth,Yheight)
          ;(X0,Yheight+1)-------(X0,Yheight+1)-------(Xwidth,Yheight+1)
          ;
```

```
                ;
                ; Steps for drawing frame:
                ;
                ;  a) Draw (X0,Y0) to (Xwidth,Y0)
                ;  b) Draw (X0,Y1),(Xwidth,Y1) to (X0,Yheight),(Xwidth,Yheight)
                ;  c) Draw (X0,Yheight+1) to (Xwidth,Yheight+1)
                ;
                ;
                ; Draw (X0,Y0) to (Xwidth,Y0)
                ;
                ;
                ;     Regs H,L = points to (X0,Y0)
                ;     Regs D,E = points to (Xx,Y0)
                ;     Reg B    = holds frame width
                ;     Reg A    = holds graphics character
                ;

                            DrawFrame:

B036 2AB4B1                  LHLD X0Y0
B039 54                      MOV D,H
B03A 5D                      MOV E,L
B03B 3AB6B1                  LDA Width
B03E 47                      MOV B,A
B03F 3AB8B1                  LDA GraphicsChr
B042 12              TopHor: STAX D ; draw (Xx,Y0)
B043 13                      INX D
B044 05                      DCR B
B045 C242B0                  JNZ TopHor

                ;
                ; Draw (X0,Y1),(Xwidth,Y1) to (X0,Yheight),(Xwidth,Yheight)
                ;
                ;
                ;     Regs H,L = points to (X0,Yy1)
                ;     Regs D,E = points to (Xx,Yy)
                ;     Reg C    = holds frame height
                ;     Reg B    = holds number of spaces between (X0,Yy1) and
                ;                (Xwidth,Yy2)
                ;     Reg A    = holds graphics character for (X0,Yy1) and
                ;                (Xwidth,Yy2) otherwise, holds the ascii code
                ;                for space (20H)
                ;

B048 3AB7B1                  LDA Height
B04B 4F                      MOV C,A
B04C 118000           Verts: LXI D,128 ; line length on OSBORNE I screen.
B04F 19                      DAD D ; point to X0,Yy1
B050 54                      MOV D,H
B051 5D                      MOV E,L
B052 3AB6B1                  LDA Width
B055 3D                      DCR A ; # spaces = Width-2
B056 3D                      DCR A
B057 47                      MOV B,A
B058 3AB8B1                  LDA GraphicsChr
```

```
B05B 12                 STAX D ; draw (X0,Yy1)
B05C 13                 INX D
B05D 3E20               MVI A,Space
B05F 12         OneVert: STAX D ; insert spaces
B060 13                 INX D
B061 05                 DCR B
B062 C25FB0             JNZ OneVert
B065 3AB8B1             LDA GraphicsChr
B068 12                 STAX D ; draw (Xwidth,Yy2)
B069 0D                 DCR C
B06A C24CB0             JNZ Verts

          ;
          ; Draw (X0,Yheight+1) to (Xwidth,Yheight+1)
          ;
          ;
          ;     Regs H,L = points to (X0,Yheight+1)
          ;     Regs D,E = point to (Xx,Yheight+1)
          ;     Reg B    = holds frame width
          ;     Reg A    = holds graphics character
          ;

B06D 118000             LXI D,128
B070 19                 DAD D ; point to (X0,Yheight+1)
B071 54                 MOV D,H
B072 5D                 MOV E,L
B073 3AB6B1             LDA Width
B076 47                 MOV B,A
B077 3AB8B1             LDA GraphicsChr
B07A 12         BotHor: STAX D ; draw (Xx,Yheight+1)
B07B 13                 INX D
B07C 05                 DCR B
B07D C27AB0             JNZ BotHor
B080 CDF3B0             CALL GraphicsOff ; disable graphics mode.
B083 C9                 RET ; exit DrawFrame

          ;
          ; DrawLogo...subroutine to draw "PU8610" on the OSBORNE I
          ;            screen when the program is in the main menu sec-
          ;            tion. Each character of the logo is configured as a
          ;            5 characters wide by a 5 line high matrix where ea-
          ;            ch segment is either a space or a special graphics
          ;            character (7FH). The graphics character appears on
          ;            the screen as a half intensity dotted rectangle.
          ;            In the following illustration the graphics charac-
          ;            ter has been replaced by an "*".
          ;
          ;            The logo "PU8610" will appear on the screen as
          ;            follows:
          ;
          ;            ****   *   *   ***    ***     *     ***
          ;            *   *  *   *  *   *  *       **    *   *
          ;            ****   *   *  *****  ****     *    *   *
          ;            *      *   *  *   *  *   *    *    *   *
          ;            *       ***    ***    ***    ***    ***
```

```
                ;
                ;           The matrix for the letter "P" is as follows:
                ;
                ;
                ;           [1,1] -->  ****  <-- [1,5]   7F,7F,7F,7F,20
                ;           [2,1] -->  *   * <-- [2,5]   7F,20,20,20,7F
                ;           [3,1] -->  ****  <-- [3,5]   7F,7F,7F,7F,20
                ;           [4,1] -->  *     <-- [4,5]   7F,20,20,20,20
                ;           [5,1] -->  *     <-- [5,5]   7F,20,20,20,20
                ;

                            DrawLogo:

B084 2A12B1                  LHLD Character1 ; set up parameters for "P"
B087 44                      MOV B,H
B088 4D                      MOV C,L
B089 1114B1                  LXI D,Character1+2
B08C CDC7B0                  CALL DrawCharacter ; go draw "P"
B08F 2A2DB1                  LHLD Character2 ; set up parameters for "U"
B092 44                      MOV B,H
B093 4D                      MOV C,L
B094 112FB1                  LXI D,Character2+2
B097 CDC7B0                  CALL DrawCharacter ; go draw "U"
B09A 2A48B1                  LHLD Character3 ; set up parameters for "8"
B09D 44                      MOV B,H
B09E 4D                      MOV C,L
B09F 114AB1                  LXI D,Character3+2
B0A2 CDC7B0                  CALL DrawCharacter ; go draw "8"
B0A5 2A63B1                  LHLD Character4 ; set up parameters for "6"
B0A8 44                      MOV B,H
B0A9 4D                      MOV C,L
B0AA 1165B1                  LXI D,Character4+2
B0AD CDC7B0                  CALL DrawCharacter ; go draw "6"
B0B0 2A7EB144                LHLD Character5 ; set up parameters for "1"
B0B3 44                      MOV B,H
B0B4 4D                      MOV C,L
B0B5 1180B1                  LXI D,Character5+2
B0B8 CDC7B0                  CALL DrawCharacter ; go draw "1"
B0BB 2A99B1                  LHLD Character6 ; set up parameters for "0"
B0BE 44                      MOV B,H
B0BF 4D                      MOV C,L
B0C0 119BB1                  LXI D,Character6+2
B0C3 CDC7B0                  CALL DrawCharacter ; go draw "0"
B0C6 C9                      RET ; exit DrawLogo

                ;
                ; DrawCharacter...subroutine to draw the letter specified by
                ;                 the parameters given in DrawLogo.
                ;
                ;
                ;                 Regs H,L = points to the segment [x,y], i.e.
                ;
                ;                     H,L = 0505H = [1,1]
                ;                     H,L = 0504H = [1,2]
                ;                           |  =  |
```

```
                ;                   H,L = 0301H = [3,5]
                ;                         |   =   |
                ;                   H,L = 0101H = [5,5]
                ;               Regs D,E = points to the contents of [x,y]
                ;                          (7FH or 20H)
                ;               Regs B,C = points to the screen memory loca-
                ;                          tion where the contents of [x,y]
                ;                          are to be stored.
                ;

                        DrawCharacter:

B0C7 210505             LXI H,0505H
B0CA 1A         Draw:   LDAX D
B0CB 02                 STAX B
B0CC 03                 INX B
B0CD 13                 INX D
B0CE 2D                 DCR L ; finished line(x)?
B0CF C2CAB0             JNZ Draw ; no, continue line
B0D2 E5                 PUSH H ; yes, go to next line
B0D3 60                 MOV H,B
B0D4 69                 MOV L,C
B0D5 017B00             LXI B,123 ; (128-5)
B0D8 09                 DAD B ; point to next line(x)
B0D9 44                 MOV B,H
B0DA 4D                 MOV C,L
B0DB E1                 POP H
B0DC 2E05               MVI L,5
B0DE 25                 DCR H ; finished drawing matrix?
B0DF C2CAB0             JNZ Draw ; no, continue
B0E2 C9                 RET ; yes, exit DrawCharacter

        ;
        ; Coutch...subroutine to send one ascii character to the
        ;          OSBORNE I screen.
        ;
        ;          uses CP/M function 2 - CONSOLE OUTPUT
        ;
        ;          reg. E = ascii character
        ;          reg. C = 02
        ;

                        Coutch:

B0E3 C5                 PUSH B
B0E4 D5                 PUSH D
B0E5 E5                 PUSH H
B0E6 47                 MOV B,A
B0E7 C5                 PUSH B
B0E8 0E02               MVI C,PrintCharacter
B0EA CD0500             CALL Bdos
B0ED C1                 POP B
B0EE 78                 MOV A,B
B0EF E1                 POP H
B0F0 D1                 POP D
```

```
B0F1 C1                 POP B
B0F2 C9                 RET

                ;
                ; GraphicsOff...subroutine to disable the graphics mode on
                ;                the OSBORNE I screen.
                ;
                ;                hex sequence = 1BH, 47H
                ;

                              GraphicsOff:

B0F3 D5                  PUSH D
B0F4 1E1B                MVI E,Esc
B0F6 CDE3B0              CALL Coutch
B0F9 1E47                MVI E,Goff
B0FB CDE3B0              CALL Coutch
B0FE D1                  POP D
B0FF C9                  RET

                ;
                ; GraphicsOn...subroutine to enable the graphics mode on
                ;                the OSBORNE I screen.
                ;
                ;                hex sequence = 1BH, 67H
                ;

                              GraphicsOn:

B100 D5                  PUSH D
B101 1E1B                MVI E,Esc
B103 CDE3B0              CALL Coutch
B106 1E67                MVI E,Gon
B108 CDE3B0              CALL Coutch
B10B D1                  POP D
B10C C9                  RET

                ;
                ; NewScreen...subroutine to clear the OSBORNE I screen
                ;                and home the cursor.
                ;
                ;                hex sequence = 1AH
                ;

                              NewScreen:

B10D 1E1A                MVI E,ClearHome
B10F C3E3B0              JMP Coutch

                ;
                ; Matrix for the letter "P" in the logo.
                ;

B112 82F1           Character1 DW 0F182H
```

```
B114 7F7F7F7F20          DB 7FH,7FH,7FH,7FH,20H
B119 7F2020207F          DB 7FH,20H,20H,20H,7FH
B11E 7F7F7F7F20          DB 7FH,7FH,7FH,7FH,20H
B123 7F20202020          DB 7FH,20H,20H,20H,20H
B128 7F20202020          DB 7FH,20H,20H,20H,20H

                ;
                ; Matrix for the letter "U" in the logo.
                ;

B12D 8AF1       Character2 DW 0F18AH

B12F 7F2020207F          DB 7FH,20H,20H,20H,7FH
B134 7F2020207F          DB 7FH,20H,20H,20H,7FH
B139 7F2020207F          DB 7FH,20H,20H,20H,7FH
B13E 7F2020207F          DB 7FH,20H,20H,20H,7FH
B143 207F7F7F20          DB 20H,7FH,7FH,7FH,20H

                ;
                ; Matrix for the number "8" in the logo.
                ;

B148 92F1       Character3 DW 0F192H

B14A 207F7F7F20          DB 20H,7FH,7FH,7FH,20H
B14F 7F2020207F          DB 7FH,20H,20H,20H,7FH
B154 7F7F7F7F7F          DB 7FH,7FH,7FH,7FH,7FH
B159 7F2020207F          DB 7FH,20H,20H,20H,7FH
B15E 207F7F7F20          DB 20H,7FH,7FH,7FH,20H

                ;
                ; Matrix for the number "6" in the logo.
                ;

B163 9AF1       Character4 DW 0F19AH

B165 207F7F7F20          DB 20H,7FH,7FH,7FH,20H
B16A 7F20202020          DB 7FH,20H,20H,20H,20H
B16F 7F7F7F7F20          DB 7FH,7FH,7FH,7FH,20H
B174 7F2020207F          DB 7FH,20H,20H,20H,7FH
B179 207F7F7F20          DB 20H,7FH,7FH,7FH,20H

                ;
                ; Matrix for the number "1" in the logo.
                ;

B17E A2F1       Character5 DW 0F1A2H

B180 20207F2020          DB 20H,20H,7FH,20H,20H
B185 207F7F2020          DB 20H,7FH,7FH,20H,20H
B18A 20207F2020          DB 20H,20H,7FH,20H,20H
B18F 20207F2020          DB 20H,20H,7FH,20H,20H
B194 207F7F7F20          DB 20H,7FH,7FH,7FH,20H
```

```
                ;
                ; Matrix for the number "0" in the logo.
                ;
B199 AAF1       Character6 DW 0F1AAH

B19B 207F7F7F20         DB 20H,7FH,7FH,7FH,20H
B1A0 7F2020207F         DB 7FH,20H,20H,20H,7FH
B1A5 7F2020207F         DB 7FH,20H,20H,20H,7FH
B1AA 7F2020207F         DB 7FH,20H,20H,20H,7FH
B1AF 207F7F7F20         DB 20H,7FH,7FH,7FH,20H

                ;
                ; Buffers used by the various routines in PU8610AS
                ;
B1B4            X0Y0 DS 2
B1B6            Width DS 1
B1B7            Height DS 1
B1B8            GraphicsChr DS 1

B1B9                    END
```